Interpret Tree Ensemble Algorithms with Purest Leaves Random Forest

Hongzhe Zhang
Email: hongzhe.zhang@pennmedicine.upenn.edu

Samprit Banerjee
Email: sab2028@med.cornell.edu

In the field of prediction modelling, one often faces a trade-off between generalization accuracy and interpretability. Random forests classifier performs well in a range of tasks owing to the power of model averaging, but the interpretability of its base learner, decision tree, is lost. In this paper, we present a novel supervised learning algorithm Purest Leaves Random Forest (PL-RF) based on a modified random forest proximity measure constructed with only the purest decision tree nodes. In this space, PL-RF performs predictions with only a small number of "relevant" decision rules therefore re-obtains the interpretability. This interpretation is demonstrated on several datasets selected from multiple science domains, and numerical experimentations suggest PL-RF has competitive performance among the similar methods and even random forests.

1 Introduction

Machine learning models are applied and being relied on in various technical fields. Their capacity to extract non-linear dependencies and high order interactions from data grants the machine learning models outstanding performances on a range of tasks of predictive modelling.[1][2] However, one often faces a trade off between predicative accuracy and interpretability. The same capacity would backfire as the extracted rules are so complicated that cannot be efficiently interpreted by a humans. The interpretability of complex machine learning models is in high needs. [3][4]

Random forest[5] as a machine learning algorithm has been known to work well in many scenarios [6] owning to the power of model averaging. Although its building blocks, decision trees, are interpretable [7], random forest itself is not. Decision tree partitions the space into a small number of sub-regions which are defined by simple, understandable rules. When a decision tree makes prediction for a new observations, the rationale behind this prediction is naturally the decision rule that defined the sub-region to which the new observation belongs. This interpretability is lost when random forest incorporates all of the trees to make a single decision, the sub-regions overlap and the number of the engaged regions explodes. The resulted decision rule would not be easily understood by a human being.

Many works have been done in order to make the random forest human-interpretable. The most practiced way is to select only a small part of the nodes generated by decision trees ensembles. As each node is represented by only one decision rule, it facilitates interpretability. Satoshi Hara Et al. [8] formed and solved this as a Bayesian model selection [9] problem. Rule ensembles [10] and Node harvest [11] treat each node as a binary indicator variable and use the linear combination of all nodes to perform prediction. To facilitate the sparsity of the coefficients and thus the interpretability, Rule ensembles constraints the 1-norm of the coefficients, while Node harvest imposes regularization by requiring that predictions are weighted node means. The inTrees [12] framework selects and combines "high quality" short rules with high prediction accuracy and frequency. The other attempts include developing a single decision tree too mimic the entire tree ensemble [13], searching for prototypical observations [14], partial dependence plots [15] and the well known variable importance [16].

In this paper, we present a new algorithm Purest Leaves Random Forest(RL-RF). Not like the works mentioned above, we obtain the interpretability not by performing post hoc analysis on random forest results. Rather, we propose a new weak learner to replace decision tree and directly aids interpretation of the tree ensembles results. RL-RF provides observation specific, decision-tree-like decision rules when performing predictions, while maintaining high, closed to random forest prediction accuracy on tested data sets. This is achieved by using only the nodes of decision trees with the highest purity in observed outcomes. We use a generalization of the tree proximity measure [17] called leave proximity. Optionally, one can select a number of prototypes using k-medoids [18] algorithm or class-specific prototype selection methods[14] with leave proximity to encourage interpretability. Predicted outcomes and observation specific decision rules can be inferred using the closest observation or selected prototype. One can also use "rule importance plot" introduced in the XXX section to summarize dominant

decision rules in each outcome.

2 Background and Notation

In this section, we introduce some notations and technical backgrounds which would be useful in later sections. Suppose we have a data set consist of N observations, each of them has p inputs and a categorical response with categories $1,2,...,K$: that is (x_i, y_i) for $i = 1,2,3,...,N$, with $x_i = (x_{i1}, x_{i2}, ..., x_{ip})$, $y_i \in \{1,2,...,K\}$.

2.1 Classification Tree

Decision tree aims to divide the feature space R^p into M sub-regions $R_1, R_2, ..., R_M$, we also call these regions nodes. And we model the response in each region as a constant c_M

$$f(x) = \Sigma_{m=1}^{M} c_M I(x \in R_M)$$

It is worth mentioning that each of the sub-region takes the form of conjunctive rule, which is easily interpretable. This serves as a important piece for the interpretability for our proposed model. If we define S_j as the set of all possible values of input variable $x_j, x_j \in S_j$, and let s_{mj} be a value from S_j, we can write sub-region R_m as

$$R_m = \prod_{j=1}^{n} I(x_j < s_{mj})$$

In each node m, representing a region R_m contains N_m observations, let

$$\hat{p}_{mk} = \frac{1}{N_m} \Sigma_{x_i \in R_m} I(y_i = k)$$

be the proportion of class k observations in node m. We classify the observations in node m to class $k(m) = \arg max_k \hat{p}_{mk}$ In this paper, we use the R package "rpart" [19] to implement the classification tree.

2.2 Random Forest

Random forest are essentially a averaged collection of "decorrelated trees". To grow a random forest, for each iteration, a decision tree is grown using bootstrapped data with only a part of variables from training data. Suppose the collection has T trees, to make a prediction at a new point x^*, we have

$$\hat{C}_{rf}^{T}(x^*) = majority\ vote\{\hat{C}_{tree}^{t}(x^*)\}$$

where $\hat{C}_{rf}$ is the prediction of random forest, $\hat{C}_{tree}^{t}$ is the prediction of the t_{th} decision tree.

2.3 Random Forest Proximity

Random forest gives a natural distance measure for observations from the perspective of each decision tree. The observations that end up in the same node are close to each other. With a random forest denoted by a collection of decision trees $\{tree_t\}_1^T$, the random forest proximity [17] is defined as

$$proximity_{RF}(x_i, x_j) = \frac{1}{T} \Sigma_{t=1}^{T} \Sigma_{m=1}^{M} I(x_i \in R_{tm}) I(x_j \in R_{tm})$$

2.4 Nearest-Neighbor Classifier

Suppose we have new unlabelled observations X^*. Nearest-neighbor Classifier[20] uses those observations in training data to construct an input space. In this space, the labels of the closest observations to X^*, are used to form the predictions $\hat{Y}^*$. Specifically, the k-nearest neighbor fit for $\hat{Y}^*$ is defined as

$$\hat{Y}(x) = \frac{1}{k} \Sigma_{x_i \in N_k(x)} y_i$$

2.5 Multidimensional Scaling

Multidimensional scaling (MDS)[21] is a method which uses distance between observations to produce a spatial representation of these observations. MDS can also be considered as a dimension reduction technique which provides a representation of the data points in low dimension (typically 2) to preserve the "configuration" of the data in higher dimension (pairwise distance $N \times N$). In mathematical terms, consider observations $x_1, ..., x_N \in \mathbb{R}^p$, denote distance between observation i and j by d_{ij}. MDS seeks $z_1, ..., z_N \in \mathbb{R}^k$ to minimize

$$\Sigma_{i \neq i'} (d_{ii'} - ||z_i - z_{i'}||)^2$$

3 Method

In this section, we introduce our novel algorithm, purest leaves. We firstly give the definition of the "purest leave" and the leave proximity measure, followed by the explanation of how purest leave and leave proximity regain the interpretability lost by random forests.

3.1 Definitions

The Purest Leave Recall the notation we gave in section 3.1, and we augment the current notation by adding the class label of the sub-region. A decision tree now contains M subregions $R_{k,m}, k \in \{1,..,K\}, m \in \{1,...,M\}$. Suppose we have a collection of T decision trees, then the purest leave of the t_{th} decision tree and class k is defined by

$$R_{k,purest}^{t} = R_{k,\arg\max_m \hat{p}_{mk}}^{t}$$

That is, the class k nodes in t_{th} decision tree with the lowest misclassification rate.

Leave Proximity We modify the random forest proximity and define the leave proximity as follows.

$$proximity_{leave}(x_i,x_j) = \frac{1}{T}\Sigma_{t=1}^{T}\Sigma_{k=1}^{K}\Big(I(x_i \in R^t_{k,purest})I(x_j \in R^t_{k,purest}) + \frac{1}{2\times K}I(x_i \notin R^t_{k,purest})I(x_j \notin R^t_{k,purest})\Big)$$

Leave proximity only uses the information in the purest leaves. When two observations are present in the same purest leave, they are naturally "close" to each other (proximity 1). If one observation falls into a purest leaf and the other does not, they are naturally distant to each other (proximity 0). When neither of two observations fall into any of the purest leaves, as the proximity only considers the information in the purest leaves, the ignorance is expressed by stating their proximity is $\frac{1}{2}$. We define the corresponding leave distance as

$$dist_{leave}(x_i,x_j) = 1 - proximity_{leave}(x_i,x_j)$$

3.2 Purest Leave Algorithm

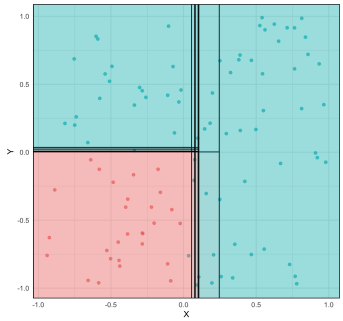
Traditional tree ensembles such as random forest cannot preserve interpretability of decision trees.

When giving prediction for a single point, the nodes of every decision trees the point falls into participate making its prediction. Sub-regions of the decision trees naturally intersect and become redundant, and some nodes maybe unstable. We use simulations to show simple examples of what we just described.

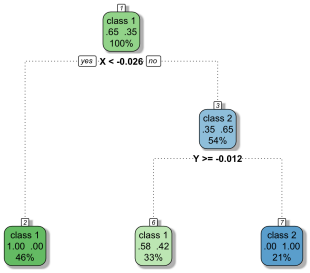
In Figure 1 (a), we simulate a two dimensional data from $uniform(-1,1)$ and assign the observations to red if both of the variables in the data are less than 0. A random forest with ten decision trees are fitted, and the feature space are girded to display the sub-regions of each decision tree. Although the random forest predictions would be perfect, the underlying information about decision rules inferred by decision trees is lost. Although the signal in the training data is simple, each decision tree learns a unique set of sub-regions and destroys the interpretability.

We again use a simulation to demonstrate the possible uncertain nodes in a decision tree. A two dimensional data is simulated from $uniform(-1,1)$, and the labels are assigned by

if $X_i \geq 0$ **then**
  $label_i \leftarrow class1$
else if $Y_i \geq 0$ **then**
  $label_i \leftarrow class1$
else
  $label_i$ *is randomly given*
end if



(a) Redundant Sub-regions of Random Forest



(b) Uncertain Node of a Decision Tree

Fig. 1

A random forest is fitted to this data, and one of the decision trees is displayed in Figure 1(b). We can see although the decision tree correctly states that the class of the data in node 6 is unclear, node 6 would still participates in the decision making of a random forest.

We believe that one can obtain interpretability by improving upon these two shortcomings of random forest. We firstly propose a novel base learner, the Purest Leave algorithm which is stated in Algorithm 1. This is an extension to decision tree, where we retain only the information in the "purest leave". This action will facilitate interpretation when the many of this base learner are ensembled. Secondly, we later show in the results section how similar and redundant rules can be seamlessly summarized for a clear interpretation.

Algorithm 1: Purest Leave Algorithm

Input: A set of points $X_{N\times p}$ and their categorical responses $Y_{N\times 1} \in \{1,...,K\}$

1 Fit a decision tree $\hat{C}_{tree}$ on a proportion of training data $Z^t_{prop*N\times q}$ bootstrapped from $X_{N\times p}$;

Output: $R_{k,purest}, k = 1,...,K$

It is obvious that the purest leaves are even weaker prediction models than decision trees. A single purest leave is guaranteed to have no knowledge to the part of feature space not covered by the purest nodes. We propose a direct application of it, called "Purest Leave Random Forest"(PL-RF). We define $k(m)$ as the majority class of sub-region m, and the steps to fit a PL-RF are given in Algorithm 2. The ensemble greatly improves the purest leave's capacity to extract information from the data. As each of the purest leaves is defined

by a single conjunctive rules, each observation in the training data will be associated with a (possibly empty) set of rules. With the results of PL-RF, we can construct a "Purest Leave" space defined by leave proximity where the data points are placed by their associated decision rules.

For new observations, predictions can be done in a k-nearest-neighbor fashion. As only a small number of training data (small k, we used 1) would participate in making the predictions, the number of decision rules associated with the prediction (which are the rules associated with the) neighbor, is significantly abated and therefore become human-interpretable (see Algorithm 3). One can perform prototype selection in the purest leaves space and reduce the decision rules to only the decision rules associated with the prototypes.We demonstrate this technique in the xxx section.

Algorithm 2: Purest Leave Random Forest

Input: A set of points $X_{N\times p}$ and their categorical responses $Y_{N\times 1} \in \{1,...,K\}$

for t = *1, ..., T* **do**
  Fit a purest leave model and log $R^t_{k,purest}$;
  for $\{i,\ X_i \in R^t_{purest}\}$ **do**
    if $Y_i = k(m)$ **then**
      Add R^t_{purest} to the set of conjunctive rules $\{Rules\}_i$
    else
      NULL
    end
  end
end

Output: Fitted purest leave model $\hat{C}^T_{leave}$, which consists of the set of purest leave $R^t_{purest}, t = 1,...,T$ and the set of decision rules $\{Rules_i, i = 1,...,N\}$

Algorithm 3: Prediction with Purest Leave Random Forest

Input: A set of unlabelled points $X^{new}_{M\times p}$, fitted purest leave model $\hat{C}^T_{leave}$ and training data $X_{N\times p}, Y_{N\times 1}$ of $\hat{C}^T_{leave}$

Let v be a $p\times 1$ vector, define $\arg min[k]v$ to be the k smallest elements of v;

for i = *1, ..., M* **do**
  1. Construct the leave distance between X_i and training data $X_{N\times p}$, denote it as $dist^{leave}_{N\times 1}$,;
  2. Log the predicted label $Y_i = majority\ vote\{Y_{\arg min[k]dist^{leave}}\}$;
  3. Log the decision rules associated $\{Rules\}_i, i \in \arg min[k]dist^{leave}$;
end

Output: Predicted labels $Y_{N\times 1}$, Associated decision rules $\{Rules\}_i, i = 1,...,N$

4 Results

4.1 Interpretation

The ultimate goal of using a interpretable model is to better understand the data, and make an astute decision based on the insights on the model [18]. Within the group of similar methods whose interpretability lies inferred interpretable decision rules; InTrees summarizes pruned decision rules which may not be entirely consistent with the data itself, NodeHarvest often results shallow, but too many decision rules to be interpreted. DegragTrees unlike tree and tree ensemble relies on probabilistic assumptions due to their Bayesian approach. Moreover, its optimization coincides with a sub-optimal solution and does not give a guaranteed convergence.

We claim that our method can better attain interpretability by constructing a PL space using the training data. New observations can be labelled by its nearest neighbors, and we can naturally claim that the predictions are induced by the rules associated with the neighbors. The space can be visualized using multidimensional scaling (MDS). We showcase the interpretation in the context of mobile health, healthcare and image analysis.

Mobile Health Data We fit the PL-RF to the smartphone data from Weill Cornell ALACRITY Center. This data set, for each subject, contains passively collected variables such as "step counts", "time away from home", and self-reported variables by subjects such as "stress". Subjects are asked everyday whether they complete the "homework" assigned by doctors to aid their depressions. Our objective is to predict homework status using passive and self-reported variables prior to the time subjects answer the "homework" question. The modelling results for one subject is shown below in Figure 2 and Table 1.

We split the 78 recorded days of the patient in to training (80%) and testing data. we used the first quantile, third quantile, median, minimun, maximum, skrewness, kurtosis and correlation with time of the prior to a day as features to predict the homework status (Yes/No) of that day. We used 50 purest leaves, 1-nearest neighbor in PL space and every training points as a medoids. For this split, we achieved a testing accuracy of 1. We show all days in Figure 2 using MDS coordinates constructed using purest leave proximity. Training data are shown by grey texts of "Yes / No" and testing data are shown by "DAY X" colored by their labels. Overlapping training and testing data are removed for a clearer presentation. The inferred decision rules that occurred more than once of "Day 9", "Day 13", "Day 18" and "Day 20" are shown in table 1. Similar rules only differ slightly in cutoffs are "averaged", as the differences were likely caused by perturbations in bootstrapped training data. We neglect the frequency of the rules in this example, but one can certainly use the frequency to represent the importance of each decision rule.

We also demonstrate interpreting a PL-RF with the "rule importance". The rule importance is simply the frequency of a rule appears in the medoids of PL-RF for each class. One can divide scale the importance of each rule by dividing its

Table 1: Explicit Interpretation for Test Observations

Day 9	Day 13
screen unlocks kurtosis $<$ -1.06	pain linear rend $<$ -0.01
time at home max $>=$ 42341.27	screen unlocks kurtosis $<$ -1.06
	step count third quantile $<$ 2450.65
	step count third quantile $>=$ 2363.07
	step count var $<$ 914672.95
Day 18	**Day 20**
step count max $<$ 2754.34	screen unlocks kurtosis $<$ -1.06
step count third quantile $<$ 2402.84	step count max $<$ 2754.34
step count var $<$ 889433.11	step count third quantile $>=$ 2363.07
tic voiced time linear trend $>=$ -53.83	step count var $<$ 888829.99
time at home max $<$ 42341.27	time at home max $<$ 42341.27

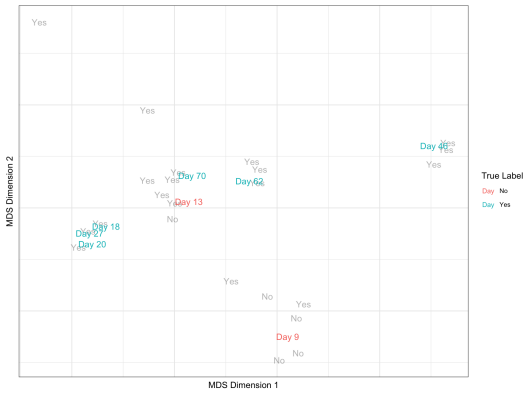


Fig. 2

frequency by the the highest frequency in its class.

Intubation Prediction From early March through mid-May 2020, the COVID-19 pandemic overwhelmed hospitals in New York City. Accurate prognostic tools to predict clinical deterioration can greatly aid the management of hospital resources including ventilators. We built a PL-RF with 50 purest leaves to predict intubation status on a cohort of COVID-19 adult patients admitted to two New York Presbyterian hospitals. Patient information includes tabular features such as their demographic characteristics, comorbidities, etc., also patients' continuously monitored labs and vitals. A sequence of well-thought representational features were extracted from lab/vital trajectories. The balanced accuracy computed with a 20% held-out test data is 0.84, and the rule importance was shown in Figure 3.

MINIST We fit the purest leave forest to sampled digits "2" and "9" from the MINIST dataset. We used 200 purest leaves to construct the leave proximity metrics, and selected 20 medoids. We again visualized it using MDS [21] on the left of Figure 3. Each of the displayed image is placed at the

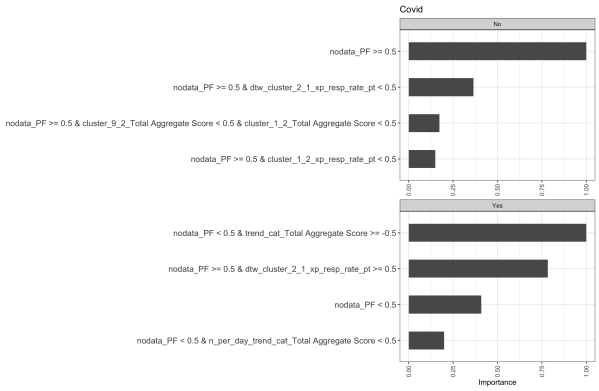


Fig. 3

position of a medoid. They are artificial images crafted as the mean of all the images in the data that satisfy the decision rules associated with that medoid. We see that 20 medoids have caused an over representation of the images at the two corners. We then reduced the number of medoids to 8, the plot of the right of Figure 2 shows that the displayed images roughly capture the shapes of different "2" and "9". This suggests that throwing away more than half of the medoids and therefore more than half of the inferred rules, does not lead to a severe loss of mined information. And the remained medoids are prototypical observations among all the digits.

4.2 Numerical Experimentation

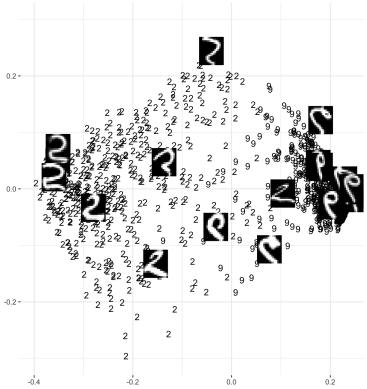
We tested the predictive performance qualitatively and compared it with other competitive methods. These include widely used random fores and decision tree, as well as similar methods Tree Space Prototypes, inTrees and Node Harvest. DefragTrees is not included for its frequently encountered convergence issue.

Datasets We picked representative datasets in the critical fields where classification algorithms with interpretations are needed. For image analysis, we picked MINIST dataset and ImageNet which are commonly used image database to benchmark the performance of a image classifier. We sampled two categories which are "cat" and "dog" from ImageNet, and digits "2" and "9" from the MINIST dataset. For healthcare, we picked mHealth data introduced in the previous section and Pima Indians diabetes database. The Boston housing dataset and glass identification dataset from USA forensic science service were picked for the section of social science.

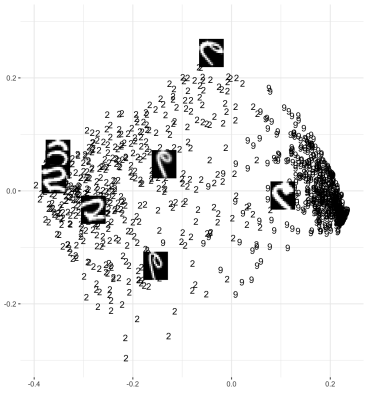
Implementation Details We extracted deep features from a pre-trained ResNet-50 for ImageNet, and used raw features for all other datasets. Decision tree, random forest (RF), Node Harvest(NH) and inTrees (IT) were implemented with the R packages "rpart", "randomForest", "nodeHarvest" and "inTrees" respectively. As there is no open-sourced code for Tree Space Prototypes (TSP), we coded the 1-NN TSP ourselves. Multi-class proximity was applied for PL-RF on all datasets. The depth is fixed to be three for decision tree. The

Model	MINIST(**3**)	ImageNet(**3**)	mHealth(**4**)	Diabetes(**3**)	Boston(**5**)	Glass(**1**)	Hepatitis
PL-RF	0.97	0.97	0.94	0.72	0.80	0.90	0.78
PL-RF-SMA	0.95	0.96	0.76	0.55	0.71	0.80	0.75
PL-RF-SMWA	0.96	0.96	0.76	0.67	0.77	0.78	NA
RF	0.99	0.98	0.96	0.74	0.86	0.90	0.82
Tree	0.93	0.90	0.50	0.71	0.82	0.53	0.73
NH	0.95	0.97	0.96	0.80	0.50	0.83	0.65
inTrees	0.95	0.50	1.00	0.68	0.84	0.84	NA
TSP SMA	0.99	0.98	0.85	0.73	0.83	0.87	0.77
TSP 1-NN	0.99	1.00	1.00	0.72	0.85	0.90	0.67

Table 2



(a) Too many (20) medoids leads to inefficient interpretation



(b) Less (8) managed to retain most of the decision information

Fig. 4

number of trees is fixed to be 50 while the number of rules for NH is chosen as number of trees $\times 5$. The number of parameter considered for each tree is fixed as $\sqrt{p}$ for TSP, RF, IT and PL-RF.

Evaluation Results In Table 2, we presented the test accuracy calculated with 20% held-out data for all the methods and datasets mentioned above. Since many of the datasets are imbalanced, we use balanced accuracy [22] as the performance metric. The rank of PL-RF out of six methods is stated after the name of each dataset. We see the generalization accuracy of PL-RF is most frequently the top 3 method and consistently superior than decision trees. It is also worth mentioning that PL-RF never has a balanced accuracy of 0.5 as owning to the multi-class proximity.

5 Concluding Remarks

We have proposed a model whose prediction and interpretation rely in the distances functions derived from its base learner. Our study suggests it can provide explicit decision rules, and has competitive performance for a variety of prediction tasks. Besides random forest, we believe the base leaner purest leave and the idea of constructing a purest leave space can be utilized by other tree ensemble methods (Eg. Random Survival Forest) to obtain interpretability.

On the other hand, the theoretical properties of purest leave random forest, like many other tree ensemble methods, remains unclear. Furthermore, as we see its generalization accuracy fluctuates for different tasks, the exact factors that determine its predicative power is unknown. Theoretical or simulation studies on PL-RF would be extremely useful in uncovering these mysteries.

References

[1] Q. Yang and X. Wu, "10 challenging problems in data mining research", International Journal of Information Technology and Decision Making, vol. 5, no. 4, pp. 597–604, 2006.

[2] X. Wu, V. Kumar, J. R. Quinlan, J. Ghosh, Q. Yang, H. Motoda, G. J. McLachlan, A. F. M. Ng, B. Liu, P. S. Yu, Z.-H. Zhou, M. Steinbach, D. J. Hand, and D. Steinberg,

"Top 10 algorithms in data mining", Knowl. Inf. Syst., vol. 14, no. 1, pp. 1–37, 2008.

[3] B. Kim, D. M. Malioutov, and K. R. Varshney. Proceedings of the 2016 ICML workshop on human interpretability in machine learning. arXiv preprint arXiv:1607.02531, 2016.

[4] A. G. Wilson, B. Kim, and W. Herlands. Proceedings of NIPS 2016 workshop on interpretable machine learning for complex systems. arXiv preprint arXiv:1611.09139, 2016.

[5] Breiman, L. Random Forests. Machine Learning 45, 5–32 (2001). https://doi.org/10.1023/A:1010933404324

[6] Rich Caruana and Alexandru Niculescu-Mizil. 2006. An empirical comparison of supervised learning algorithms. In Proceedings of the 23rd international conference on Machine learning (ICML '06). Association for Computing Machinery, New York, NY, USA, 161–168. DOI:https://doi.org/10.1145/1143844.1143865

[7] Alex A. Freitas. 2014. Comprehensible classification models: a position paper. SIGKDD Explor. Newsl. 15, 1 (June 2013), 1–10. DOI:https://doi.org/10.1145/2594473.2594475

[8] Hara, S., Hayashi, K. (2018). Making Tree Ensembles Interpretable: A Bayesian Model Selection Approach. AISTATS.

[9] R. E. Kass and A. E. Raftery. Bayes factors. Journal of the American Statistical Association, 90:773–795, 1995.

[10] Friedman, Jerome H.; Popescu, Bogdan E. Predictive learning via rule ensembles. Ann. Appl. Stat. 2 (2008), no. 3, 916–954. doi:10.1214/07-AOAS148. https://projecteuclid.org/euclid.aoas/1223908046

[11] N. Meinshausen. Node harvest. The Annals of Applied Statistics, 4(4):2049–2072, 2010.

[12] H. Deng. Interpreting tree ensembles with intrees. arXiv preprint arXiv:1408.5456, 2014.

[13] Zhou, Y., Zhou, Z., Hooker, G. (2018). Approximation Trees: Statistical Stability in Model Distillation. ArXiv, abs/1808.07573.

[14] Sarah Tan, Matvey Soloviev, Giles Hooker, Martin T. Wells Tree Space Prototypes: Another Look at Making Tree Ensembles Interpretable arXiv:1611.07115 [stat.ML]

[15] Friedman, Jerome H. Greedy function approximation: A gradient boosting machine. Ann. Statist. 29 (2001), no. 5, 1189–1232. doi:10.1214/aos/1013203451. https://projecteuclid.org/euclid.aos/1013203451

[16] Ishwaran, H. Variable importance in binary regression trees and forests. Electronic Journal of Statistics.

[17] Breiman, L., and Cutler, A. 2002. Random forests manual. https://www.stat.berkeley.edu/breiman/RandomForests.

[18] Hae-Sang Park, Chi-Hyuck Jun, A simple and fast algorithm for K-medoids clustering, Expert Systems with Applications, Volume 36, Issue 2, Part 2, 2009, Pages 3336-3341, ISSN 0957-4174, https://doi.org/10.1016/j.eswa.2008.01.039.

[19] Terry Therneau and Beth Atkinson (2019). rpart: Recursive Partitioning and Regression Trees. R package version 4.1-15. https://CRAN.R-project.org/package=rpart

[20] Hastie, T., Tibshirani, R., Friedman, J. H. (2009). The elements of statistical learning: data mining, inference, and prediction. 2nd ed. New York: Springer.

[21] Cox M., Cox T. (2008) Multidimensional Scaling. In: Handbook of Data Visualization. Springer Handbooks omp.Statistics. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-540-33037-0-14

[22] Kay Henning Brodersen, Cheng Soon Ong, Klaas Enno Stephan, and Joachim M Buhmann. 2010. The Balanced Accuracy and Its Posterior Distribution. In International Conference on Pattern Recognition.